**Recursion:**

The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called a recursive function. Using a recursive algorithm, certain problems can be solved quite easily. Examples of such problems are [Towers of Hanoi (TOH)](https://www.geeksforgeeks.org/c-program-for-tower-of-hanoi/), [Inorder/Preorder/Postorder Tree Traversals](https://www.geeksforgeeks.org/tree-traversals-inorder-preorder-and-postorder/), [DFS of Graph](https://www.geeksforgeeks.org/depth-first-traversal-for-a-graph/), etc. A recursive function solves a particular problem by calling a copy of itself and solving smaller sub problems of the original problems.

Recursion is an amazing technique with the help of which we can reduce the length of our code and make it easier to read and write. It has certain advantages over the iteration technique.A task that can be defined with its similar sub-task, recursion is one of the best solutions for it.

**Properties of Recursion:**

1. Performing the same operations multiple times with Different inputs.
2. In every step, we try smaller inputs to make the problem smaller.
3. Base condition is needed to stop the recursion otherwise infinite loop will occur.

**How are recursive functions stored in Memory?**

Recursion uses more memory, because the recursive function adds to the stack with each recursive call, and keeps the values there until the call is finished. The recursive function uses Last in First Out structure like Stack data structure.

The whole idea is to represent a problem in terms of one or more smaller problems, and add one or more base conditions that stop the recursion.

Stack Overflow can happen in case base condition is not reached or not defined.

1. Direct Recursion:Direct Recursion is the one if the function calls itself inside the function
2. Indirect Recursion: If a function calls another function which calls this function directly or indirectly.
3. Tailed Recursion: A recursive function is tail recursive when a recursive call is the last thing executed by the function.
4. Non-tailed Recursion: A recursion function is non-tail recursive when a recursive call is not the last thing executed by the function.

**How is Memory allocated to different function calls in recursion?**

When any function is called from main(), the memory is allocated to it on the stack. A recursive function calls itself, the memory for a called function is allocated on top of memory allocated to the calling function and a different copy of local variables is created for each function call. When the base case is reached, the function returns its value to the function by whom it is called and memory is de-allocated and the process continues.  
Let us take the example of how recursion works by taking a simple function.

**Recursion vs Iteration**

|  |  |
| --- | --- |
| **RECURSION** | **ITERATION** |
| Terminates when the base case becomes true. | Terminates when the condition becomes false . |
| Used with functions. | Used with Loops. |
| Every recursive call needs extra space in the Stack memory. | Every iteration does not require any extra space. |
| Smaller code size. | Larger code size. |

Problem:

1. Write a program and recurrence relation to find the Fibonacci series of n where n>2 .
2. Write a program and recurrence relation to find the Factorial of n where n>2 .

|  |  |
| --- | --- |
| **Disadvantages of Recursive Approach** | **Advantages of Recursive Approach** |
| 1. The recursive program has greater space requirements than the iterative program as all functions will remain in the stack until the base case is reached. 2. It also has greater time requirements because of function calls and returns overhead. 3. The codes are difficult to understand and hence extra care has to be practiced while writing the code. 4. The computer may run out of memory if the recursive call is not checked properly. | 1. Recursion provides a clean and simple way to write code. 2. Some of the problems like the tower of Hanoi is preferred to be written in Recursive format. |